sends a request to the Internet **610**. At **630**, upon receipt of the uplink traffic from UE**2**, the RACS **608** performs the application behavior analysis (in accordance with method **500** described above). At **632**, the Internet responds to the request from UE**2**. Upon receipt of the downlink traffic from the Internet **610**, at **634** the RACS performs application behavior analysis (in accordance with method **500** described above). At **636**, and after the completion of the application behavior analyses, the RACS **608** generates an adapted timer value and sends it to both the Policy Server **612** and the eNB **606**. At **638**, the adapted timer value is sent to both UE**1** and UE**2**, which then adopt the value.

[0034] The user device or UE **700** is illustrated in the block diagram of FIG. **7**. As shown in FIG. **7**, the UE **700** includes a processor **702** configured to communicate with a network **704**, and a memory **706** in communication with the processor. The processor **702** is configured to, in accordance with the methods **200**, **300**, **400** and **500** described above, connect the UE **700** to the network **704**, initiate traffic on the network, and adopt an adapted timer value based on the traffic initiated on the network. The UE **700** is in communication with an application server or RACS **708**, and as described in detail above with reference to methods **200**, **300**, **400** and **500**, the application server is configured to, among other things, receive a request from the UE, perform an application behavior analysis based on the request, determine an adapted timer value based on the application behavior analysis, and send the adapted timer value to the UE. Although other devices may be appropriate, the UE **700** is a communication device such as a portable communication device, mobile communication device, smartphone, tablet, laptop, or personal computer.

[0035] FIGS. **8**-**10** show the application-usage behavior of several user equipment devices UEx attached to the RAN over a period of time. Specifically, in FIG. **8**, UE**1**-UE**75** are shown and their behavior and frequency of access tracked over a period of time and utilizing a fixed timer value. As can be seen by the outputs on graph **800**, it is difficult to derive any knowledge from the outputs in graph **800** because there is no synchronization between the user accessing the system, the device generating a signaling load, and application-generating traffic. Turning next to FIG. **9**, UE**1**-**75** are again tracked and grouped together based on similar network characteristics. The UEs are tracked based on their user access throughout the day and the frequency of their access. In graph **900** shown in FIG. **9**, due to the learning process described in the present disclosure, UEs are grouped together based on access patterns and OS types of the UEs, thus rendering it possible to determine adaptive timer values that can reduce signaling load on the network. Taking the analysis one step further, FIG. **10** shows graph **1000**, in which usage/frequency of various application types is tracked based on user access throughout the day and the frequency of such access. As seen in graph **1000**, clusters of usage are clearly formed, with different colors within the clusters indicating different data/subscription plans, for example. This information, which is part of the learning process described in detail above, can be used to develop adaptive timer values that can increase the scalability of the network, for example.

[0036] The present disclosure provides an apparatus and methods for implementation of an adaptive timer value for use in Radio Resource Control. Such an adaptive timer is becoming more important because of the nature of the Internet and the applications running on user devices. For example, with the onset of more video content and chat features in Internet applications, traditional web page requests are becoming less prevalent. Accordingly, traditional fixed timer mechanisms for RRC are becoming less efficient because the various types of applications (i.e., video, IM, chat, traditional web traffic, VoIP, gaming) each have different requirements on the network. Furthermore, because user devices have different OS, models, and application versions, fixed timer configurations are unable to efficiently handle various RRC requests.

[0037] The present methods and apparatus provide a machine learning approach (using the RACS or application server) to learn and analyze various inputs including application flows, device characteristics, and network load characteristics, for example, to dynamically configure adaptive timer values. The methods disclosed herein result in conserved UE energy, improved BTS scheduling, and reduced radio signaling, among other features. In addition, the present apparatus and methods can enhance UE energy for future flows by utilizing information from each UE and by consulting an HHD that stores previously analyzed application behaviors. Further, because the adaptive timer values are UE-specific and not network specific, UE efficiency is increased and UE energy is conserved. Utilization of the present apparatus and methods can also improve radio scheduling because of the use of the adaptive timer values. Also, radio signaling messages can be reduced by implementing the present apparatus and methods, because the application server is doing the majority of the work by performing the application behavior analyses or learning method and by consulting the HHD for previously stored device/profile information.

[0038] Embodiments of the present disclosure may be implemented in software (executed by one or more processors), hardware (e.g., an application specific integrated circuit), or a combination of software and hardware. In an example embodiment, the software (e.g., application logic, an instruction set) is maintained on any one of various conventional non-transitory computer-readable media. In the context of this document, a "non-transitory computer-readable medium" may be any media or means that can contain, store, communicate, propagate or transport the instructions for use by or in connection with an instruction execution system, apparatus, or device, such as a computer. A non-transitory computer-readable medium may comprise a computer-readable storage medium (e.g., memory or other device) that may be any media or means that can contain or store the instructions for use by or in connection with an instruction execution system, apparatus, or device, such as a computer. As such, the present invention includes a computer program product comprising a computer-readable storage medium bearing computer program code embodied therein for use with a computer, the computer program code comprising code for performing any of the methods and variations thereof as previously described. Further, the present invention also includes an apparatus which comprises one or more processors, and one or more memories including computer program code, wherein the one or more memories and the computer program code are configured, with the one or more processors, to cause the apparatus to perform any of the methods and variations thereof as previously described.